MC132x

High Speed CMOS GigE Vision Camera

MC132x Users Manual Rev. 0.23
Camera-Firmware: B2.02-V2.16-F0.81
Kamera ID Nr.: MC1324 .. MC1327

Mikrotron GmbH
Landshuter Str. 20-22
D-85716 Unterschleißheim
Germany

Tel.: +49 89 726342 00
Fax: +49 89 726342 99
info@mikrotron.de
www.mikrotron.de

Table of contents

1 General

1.1 For customers in the U.S.A.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense. You are cautioned that any changes or modifications not expressly approved in this manual could void your authority to operate this equipment. The shielded interface cable recommended in this manual must be used with this equipment in order to comply with the limits for a computing device pursuant to Subpart J of Part 15 of FCC Rules.

1.2 For customers in Canada

This apparatus complies with the Class A limits for radio noise emissions set out in Radio Interference Regulations.

1.3 Pour utilisateurs au Canada

Cet appareil est conforme aux normes Classe A pour bruits radioélectriques, spécifiées dans le Règlement sur le brouillage radioélectrique.

1.4 Life Support Applications

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Mikrotron customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Mikrotron for any damages resulting from such improper use or sale.

1.5 Declaration of conformity

Manufacturer: **Mikrotron GmbH**

Address: Landshuter Str. 20-22
85716 Unterschleissheim
Deutschland

Product: **camera** **MC1324, MC1325, MC1326, MC1327**

The dedicated products conform to the requirements of the Council Directives 2004/108/EG for the approximation of the laws of the Member States relating to electromagnetic consistency. The following standards were consulted for the conformity testing with regard to electromagnetic consistency.

EC regulation	Description
EN 61000-6-3	Electromagnetic compatibility
EN 61000-6-1	Immunity

Unterschleissheim, August 04th. 2006

Mikrotron GmbH

Dipl.-Ing. Bernhard Mindermann
President of Mikrotron

1.6 Warranty Note

Do not open the body of the camera. The warranty becomes void if the body is opened.

1.7 Remarks, Warnings

This document contains important remarks and warnings. See the corresponding symbols:

	**Important remark**

	**Attention, Warning**

2 Introduction

The CMOS high speed camera MC132x is a high resolution camera with 1280 x 1024 pixel. Benefits of CMOS technology are high speed, random access to pixels with free programmability and low power.

The camera uses industry-standard C-Mount lenses. The sensor diagonal is 1,25“ with square pixels measuring 12µm.

Free programmability means that the user is free to define the region of interest by size and position and the speed of data output. The frame rate can be selected between 25 fps and several thousand fps depending on resolution and video data width.

With a resolution of 1280 x 1024 pixel, 80 fps can be output via the Gigabit Ethernet Interface. Parameter sets are called “profiles” and stored in non volatile memory.

2.1 Top level specifications

- high resolution: 1.280 x 1.024 pixel CMOS sensor
  with 1300 A/D converters
- up to 1.024 gray levels
- up to 80 full frames/s
  up to 500 frames/s using data reduction algorithms
- arbitrary region of interest
- high sensitivity
- 12µm square pixels
- electronic “Freeze Frame” shutter
- low blooming
- programmable via serial link
- patented ImageBLITZ® image trigger
- asynchronous trigger
- download customer specific FPGA preprocessing firmware
- small, compact housing
- wide power supply range

2.2 Electronic „Freeze Frame“ Shutter

Preceding exposure, the contents of all light sensitive elements is cleared. When exposure terminates, accumulated charge is transferred to an analog memory associated which each pixel. It stays there until it is read out (and discharged) by the A/D conversion cycle.

As all light sensitive elements are exposed at the same time, even fast moving objects are captured without geometric distortion.

2.3 Differences between the camera types

The CMOS cameras are available in different versions depending on the supported features monochrome/color or Base/Full Camera Link ® interface, lens connection to C-mount or changeable lens adapter C-mount/F-mount.

Features / Type	Data width (bits)	Color/ Mono	Max BWidth	C/F-Mount lens adaption	max. frame rate@ 1280 x 1024	Image-BLITZ®	Image pre-processing supported
MC1324	8	M	110 Mbyte/s	C	80 fps	+	+
MC1325	8	C	110 Mbyte/s	C	80 fps	+	+
MC1326	8	M	110 Mbyte/s	F	80 fps	+	+
MC1327	8	C	110 Mbyte/s	F	80 fps	+	+

Table 2.3-1

+ ... supported
- ... not supported

2.4 Using the camera

There are no serviceable parts inside the camera.. The camera may not be opened, otherwise guarantee is lost.

Use dry, soft lens-cleaning tissue for cleaning lenses and, if necessary, the sensors window.

3 Hardware

3.1 GigE Vision® Interface

GigE Vision® is designed for digital cameras in machine vision applications. This interface can transfer data at a rate of 110 Mbytes/sec.

The GigE Vision® chipset and software is designed and implemented by Pleora Tech. Inc. the leading supplier of GigE Vision technology. Please refer to the PT1000 documentation for further information.

3.1.1 Serial command interface

The camera is controlled by an ascii-based serial protocol, wrapped by the Pleora IPORT engine.

3.1.2 GENICAM® command interface

Implemented camera specific GENICAM® features:

Feature name	Corresponding serial command	Reference
CameraHeadSensorClock	:S4xxxxx	Clock selection
CameraHeadPixelClock	:S6xxxxx	Clock selection
CameraHeadDigitalGain	:r700x	Digital gain
CameraHeadSaveAsPowerUp	:pc	Profile processing
CameraHeadReset	:c	RESET of the camera

3.2 Power supply

The camera needs a DC supply voltage between 8 … 24 V at a power consumption of 5,0 Watt max.

See also Connector pinning .

Before applying power to the camera we strongly recommend to verify the used pins of the power connector, the polarity (+/-) of the leads and the supply voltage.

The camera may only be used with a supply voltage according to the camera specification. Connecting a lower or higher supply voltage, AC voltage, reversal polarity or using wrong pins of the power connector may damage the camera. If doing so, the warranty will expire immediately.

3.3 Status LED

A dual colour LED on the camera backplane shows the operating condition of the MC132x.

LED orange... The MC132x is configuring the internal FPGA. No other activity is possible.

LED green...	The MC132x is fully operational.
LED off...	If LED is off, despite the camera is powered, the FPGA configuration data is reloaded via the serial interface and stored in internal EEPROM. No other activity is possible.
LED red...	red LED that the FPGA could not be loaded because of wrong FPGA configuration data. Try to reload configuration data.

4 Getting started

Before starting to operate the camera, make sure that the following equipment is available:

- Camera MC132x
- C-Mount/F-Mount Lens
- Mikrotron Support CD
- Image processing system, e.g.: PC and Software

ⓘ	For GigE Interface the Intel MT1000 Chipset is recommended

Additional items:

- 1 GigE cable (CAT6 recommended)
- 1 Power supply 12VDC, 0.5A min
- 1 power cable

ⓘ	To specify cables see chapter Connector pinning.

4.1 First steps

1. Switch off the image processing system
2. Connect GigE cable between camera and PC.
3. Connect power cable.
4. Unscrew dust protection cover, screw in lens.
5. Switch on the image processing system and camera power supply

5 Initial setup

The MC132x is delivered with initial parameters and therefore does not need to be configured via the serial link.

5.1 Serial number and firmware revision

Serial number and firmware revision is provided in MC132x non volatile memory. Use **:v** command (Read serial number and firmware revision) to read serial number and firmware revision. The serial number is also marked on the type plate of the camera.

5.2 PowerUpProfile

The PowerUpProfile is the setting of all registers, which is loaded from non-volatile memory during power-up and software reset. The content of the PowerUpProfile is loaded into the Camera Profile.

5.3 Camera Profile

The actual set of parameters is called Camera Profile. All changes of parameters by the serial link is reflected in the Camera Profile. The Camera Profile can be saved to 8 user profiles or one PowerUpProfile by command. It is loaded from 8 user profiles or 8 factory profiles or the PowerUpProfile. The camera profile is volatile and must be stored to the PowerUpProfile to be reactivated on next power up or software reset.

5.4 Factory profile

The factory profile can be read but not written by the user. They are factory preset to the settings of table below.

Profil-Nr.	Resolution in pixel	Data width per Pixel in bit	Frame rate in fps	Video data width in Mbyte/s
0	100 x 100	8	4.852	48,5
1	240 x 240	8	1.011	58,2
2	640 x 480	8	202	62,1
3	1280 x 1024	8	47	61,6
4	640 x 480	8	202	62,1
5	1280 x 1024	8	47	61,6
6	640 x 480	10	48	28,1
7	1280 x 1024	10	23	57,5

Table 5-1

5.5 User profile

The user can store up to eight User profiles in non volatile memory. All load or write commands exchange data between the PowerUpProfile and one of the four user profiles. The user profiles are factory preset to the factory profiles.

6 Configuration

The MC132x has 15 FPGA registers, r1..rf_h , each 10 bit wide, 16 x 32bit registers, eight D/A registers, a1..a8, 8-bit wide, and two clock code registers, each 3 x 8bit wide. The contents of all the above registers is called a profile. There is space in non volatile memory for 17 profiles: one PowerUpProfile, 8 user profiles and 8 factory profiles.

Any change of a specific register through the serial interface is immediately processed and written to the volatile part of the memory and gets lost when power goes down. A command must be used to store the actual setting in non volatile memory. After power-up the PowerUpProfile is loaded from the non-volatile to the volatile part of the memory (Camera profile).

A load or write command exchanges data between the PowerUpProfile and one of the eight user profiles. The eight factory profiles can be read but not be written by any command. All values are given in hexadecimal notation, e.g.: 0xff or 0ffh = 255.

6.1 Commands

ASCII strings are used to change camera parameters. All commands start with a colon, followed by one selection character and a value in hexadecimal notation with two or three ASCII characters.

After a command has been recognized, processing is immediate, for all commands but the save type commands (:px). These need a EEPROM write time of app 1ms. An answer is provided with read type commans (:v, :w, :W) or, if the command “command acknowledge flag” is set, after processing of each command an ACK or NAK character. Processing of wrong command is stopped immediately on recognizing the error. A new command must start with a colon.

6.1.1 Table of commands

Syntax	Range	Answer	Description
:a<n><xx>	<n> = 1...8 <xx> = 0...ff$_h$	ACK*	Set one of eight analog voltages for the sensor
:A<n>	<n> = „y“,“Y“,“n“,“N“	ACK*	En- or disable a command acknowledge or not acknowledge (ACK or NAK)
:b<n>	<n> = 0...4	ACK*	Select baud rate: 0=9600 Bd (default setting), 1=19.2 kBd, 2=38.4 kBd, 3=56.8 kBd, 4=115.2 kBd
:c	--	ACK*	RESET and new Initialization of the camera, new load of PowerUpProfile. Duration: some seconds
:e...	--	--	Transmit & save a new FPGA configuration
:ERASE<ccc>	<ccc> = “APP”, “EPCS1”	--	Erase of camera internal firmware, **CAMERA STOPS WORKING, READ COMMAND DESCRIPTION BEFORE EXECUTING !**
:f<n>	<n> = 0...7	ACK*	Reload one of eight, factory defined and calibrated profiles to PowerUpProfile.
:g<n>	<n> = 0...7	ACK*	Reload one of eight user profiles to PowerUpProfile
:l<nn><xxxxxxxx>	<nn>=00..02 <xxxxxxxx>=32bit value	ACK*	Extended registers for special use (e.g. direct input of camera settings), 32 bit values, as 6 ASCII char
:p<n>	<n> = 0...7	ACK*	Save PowerUpProfile to one of eight user profiles, allow app. 1ms save time.
:r<n><xxx>	<n> = 1...f$_h$ <xxx>= 000...3ff	ACK*	Write a FPGA - register
:S<xxxxxx>	<xxxxxx> = 6 Byte Code	ACK*	Program sensor and pixel clock directly.
:t<n><m>	<n> = 00..7f$_h$ <m> = 00..ff$_h$	ACK*	Short setting of X- position in units of 10 pixel and Y-position in units of 4 lines.
:T	--	+50.5	Read temperature
:v	--	#12345-B2.02-V1.10-F1.29	Read serial number (#) and firmware versions
:V	--	1324000003433	Read identifier
:w	--	camera profile: 44 bytes in hex	Read actual Camera Profile, data output in hex
:W	--	Camera profile: 44 bytes in ASCII	Read actual Camera Profile, data output in ASCII
:Z<c><…>	<c> = “a”, “c”, “l”, “r” <…>= register no., depending on <c>	register value in ascii, different formats	Read single register values

ACK* … ACK (acknowledge, 06 hex) or NAK (not acknowledge, 15 hex) will only be sent if the response was enabled first by command :A<n>.

6.1.2 RESET of the camera

For reconfiguring the camera without power down/up there is a command for resetting the camera. The camera will start-up new, configure fpga and load the power-up profile. Additionally some volatile settings will be adjusted to the default values, e.g. baud rate.

Command: **:c** camera reset

Response: ACK if enabled by separate command :A<n>

6.1.3 Acknowledge response

Several commands do only adjust the camera without any response of the camera. To get an answer of the camera an acknowledge response can be en- or disabled. The default setting for camera start-up is “no response”. After power-up or reset of the camera the Acknowledge response is disabled. The response is volatile and must be separately enabled by the following command.

Command: **:A<n>** <n> = “Y”, “y”, “N”, “n”
“Y”, “y” … response enabled
“N”, “n” … response disabled

Response:	ACK	if command is “:AY” or “:Ay” ACK = 06 hex, in hex NAK = 15 hex, in hex

All commands, which can answer with a acknowledge reponse are shown in the command table.

☝	**Provided that acknowledge response is enabled, the camera will only send ACK or NAK if it receives a known command structure with at least a colon. Sending any characters without command syntax will not be responded.**

6.1.4 Baud rate

The baud rate of the serial communication may be adjusted from low to higher speed. The camera always starts with 9.600,8,n,1,- (9.600 Bd, 8 data bits, no parity, 1 stop bit). The adjustment of the baud rate can be changed, but the setting is volatile and will be reset to 9.600 Bd if power-up or reset of the camera occurs.

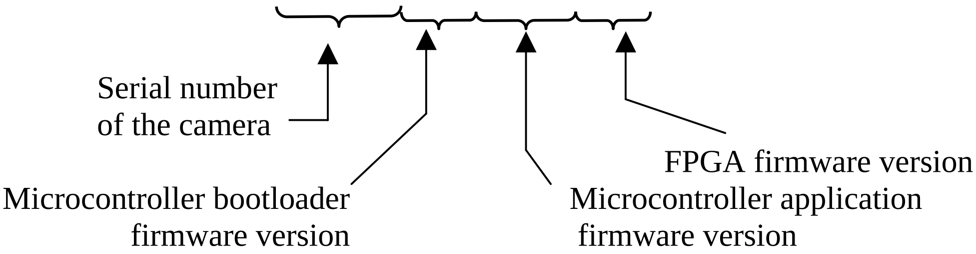
Command:	**:b<*n*>**	<*n*> = 0 ... 4 0 = 9.600 Bd (default, power-up, reset) 1 = 19.200 Bd 2 = 38.400 Bd 3 = 57.200 Bd 4 = 115.400 Bd
Response:	ACK	if enabled by separate command :A<n>

6.2 Read camera information

6.2.1 Read serial number and firmware revision

The serial number and the firmware revision can be read with the **:v** command.

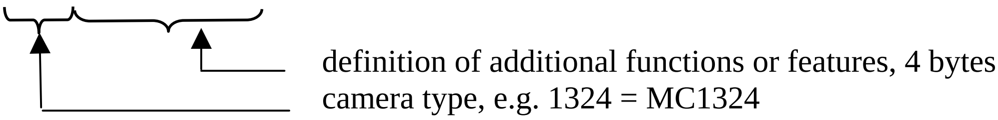
Command: **:v**
Response(e.g.): #01234-B2.02-V2.02-F0.71



6.2.2 Read identifier

The identifier offer information about the camera type and camera functions. It consists of 8 bytes, which are delivered as 16 ascii characters.

Command: **:V**
Response (e.g.): 1324000000000000



Definition of functions or features: cccc0000000000ss

00 … not defined

cccc … camera type of camera label at the rear housing of the camera, e.g. 1324, 1325, 1326, 1327

ss … sobel filter

Function	bit	value of each bit of ss in hex	description
not used	7	80	1 … enable, 0 … disable
not used	6	40	1 … enable, 0 … disable
not used	5	20	1 … enable, 0 … disable
Sobel bin+dir (2bit per pixel)	4	10	1 … enable, 0 … disable
Sobel binary (1bit per pixel)	3	08	1 … enable, 0 … disable
Sobel bin+dir (8bit per pixel)	2	04	1 … enable, 0 … disable
Sobel (8bit per pixel)	1	02	1 … enable, 0 … disable
Target data (proprietary)	0	01	1 … enable, 0 … disable

Example: Sobel filter with all features: ss = ff

6.2.3 Read complete camera settings

The complete, actual camera settings can be read out . The answer are the values of all camera registers.

Command: **:w** Output as hexadecimal digits
:W Output as ASCII-String

Example for ":w" (output as hex digits, 110 Databytes + CR + LF):

```
6d6448c66500650061008541898c0000
03f f03f f0000007f 0030000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
000000000000000000000000↵
```

15x10bit registers r1... rf

16x32bit registers m0...mf

CR+LF

Example for ":W" (output as ASCII string, 91 Bytes total, 88 databytes, 1x CR preceding the databytes, 1x CR after 32 ASCII-characters and 1x CR after 64 ASCII- characters):

```
↓6d6448c66500650061008541898c0000↓
03f f03f f0000007f 0030000000000000↓
00000000000000000000000000000000↓
00000000000000000000000000000000↓
00000000000000000000000000000000↓
00000000000000000000000000000000↓
000000000000000000000000↵
```

CR

Assignment of data to camera parameters:

analog settings

```
6d6448c665006500 61008541898c 0000
03f f03f f0000007f 0030000000000000
000000000000000000000000↵
```

image size & position

Codes for pixel- and Sensorclock

Transmitted bytes:
A1 A2 A3 A4 A5 A6 A7 A8 Sa1 Sa2 Sa3 Sb1 Sb2 Sb3
R1h R1l ... R15h R15l M0hh M0hl M0lh M0ll … Mfhh Mfhl Mflh Mfll↵

A1...A8	analog settings	
Sa1 Sa2 Sa3	3 Bytes frequency codes of pixelclock	(see 6.7)
Sb1 Sb2 Sb3	3 Byte frequency codes for sensorclock	(see 6.7)
R1…R15	image size & position (r registers)	
R1h ...	high Byte Register1	
R1l ...	low Byte Register1	
M0…Mf	fpga registers (m registers)	
M0hh	1. byte of m register no. 0, MSB	
M0hl	2. byte of m register no. 0	
M0lh	3. byte of m register no. 0	
M0ll	4. byte of m register no. 0, LSB	
↵ ...	CR+LF (0dh + 0ah)	

Abb.	description	hex. Code
CR	carriage return	0d
LF	line feed	0a

6.2.4 Read single values of camera setting

All camera internal registers of settings may also be read as single values.

Command: **:Z<c><xx>** request for single register value of camera setting
<c> … 'a', 'c', 'r', 'm' (ascii characters)
<xx> … extension, which can differ in dependence of the wanted camera setting

Valid commands:

:Za<n>	output of value of analog register, in ascii characters <n> … 1 - 8 (ascii character)
:Zcs	output of sensor clock code
:Zcp	output of pixel clock code
:Zr<xx>	output of fpga registers r, <xx> … 01 – 0f (hex, in ascii characters)
:Zm<n>	output of special fpga registers m, <n> … 0 – f (hex, in ascii characters)

Response:

for :Za<n>	, e.g. 6d↓	(hex, 2 ascii characters + CR)
for :Zcs	, e.g. 41bb0b↓	(hex, 6 ascii characters + CR)
for :Zr<xx>	, e.g. 03ff↓	(hex, 4 ascii characters + CR)
for :Zm<n>	, e.g. 00000000↓	(hex, 8 ascii characters + CR)

or for incomplete or invalid command (e.g. :Zx): NAK if enabled by separate command :A<n>

6.2.5 Reading camera temperature

To control the temperature inside , the camera disposes of an internal temperature sensor. The temperature inside the camera can be read out in steps of 0.5°. The value is delivered in ASCII characters signed.

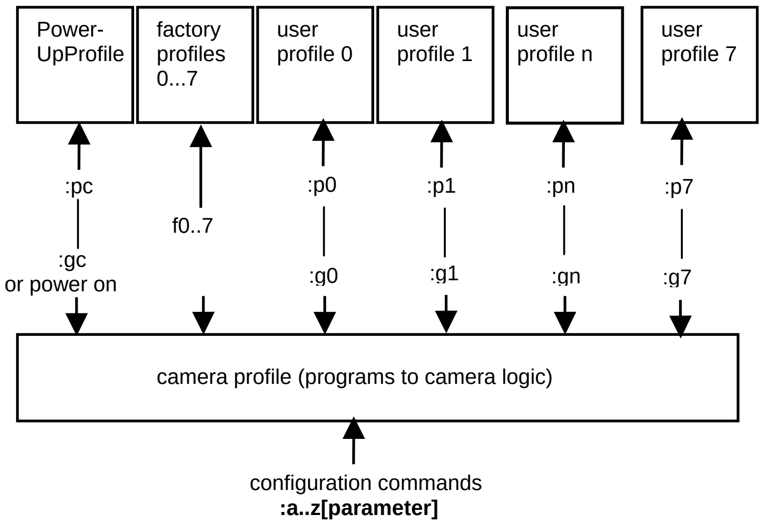
Command: **:T**

Response(e.g.): +34.0
The temperature sensor is able to deliver values of –128°C to +128°C.

	Take care that the temperature of the camera does not exceed the specified case temperature range.

6.3 Profile processing

All camera settings are loaded or stored as complete data blocks (= Profiles). There are 17 profiles, the Camera profile, the PowerUpProfile, eight factory profiles and eight user profiles.



6.3.1 Write user profile

The Camera Profile is transferred to one of the eight user profiles (0…7) or the power-up profile (c). The user profiles are located in non volatile memory.

Command:	**:p<*n*>**	<*n*> = 0 ... 7,c
Response:	ACK	if enabled by separate command :A<n>

	Issue this command only, if the Camera Profile was successfully tested.

6.3.2 Load user profile

Load one of eight user profiles or the power-up profile, which are located in non volatile memory, to the Camera Profile. Additionally all registers will be programmed with the new values.

Command:	**:g<*n*>**	<*n*> = 0 ... 7, c
Response:	ACK	if enabled by separate command :A<n>

6.3.3 Load factory profile

The eight factory profiles, located in non volatile memory, can be read but not changed by the user. With this command factory profile no. <n> is loaded into Camera Profile. Additionally all registers will be programmed with the new values.

Command:	**:f<*n*>**	<*n*> = 0 ... 7
Response:	ACK	if enabled by separate command :A<n>

6.4 Image quality

There are eight analog parameters to adjust the first stage of the sensor and influence image quality. Five of them are calibrated for getting excellent image quality and should not be changed. Three parameters are also preset for good image quality. Although it's not necessary to change them, it can be useful to optimize them depending on the used camera mode and speed. The three parameters are FPN, Gain, and Black up.

6.4.1 FPN

The **F**ixed **P**attern **N**oise setting reduces the fixed pattern noise that is typical to CMOS sensors. This level might be changed if the sensor clock frequency is changed. For adjustment set the lens out of focus and to a medium grey level. Lower FPN until a heavy pattern appears. Then raise by a few points.

Command:	**:a1<x_1x_0>**	<x_1x_0> : Range 00 .. ff (hex), typ. 55h ... 80h, ASCII char
Response:	ACK	if enabled by separate command :A<n>
Example:	:a164	set register a1 to 64 (hex)

6.4.2 Gain

This is the threshold for the A/D converters. To increase the gain the value of a2 must be lowered. Values below 30h will lead to oversensitive images and are not recommended to set.

Command:	**:a2<x_1x_0>**	<x_1x_0> : Range 00 ... ff (hex), typ. 30 ... 80 (hex), ASCII char
Response:	ACK	if enabled by separate command :A<n>
Example:	:a264	set register a2 to 64 (hex)

6.4.3 Black Level

If no Black Level is adjusted (= 00 hex), each pixel will have an offset, which reduces the contrast. For excellent image quality it's recommended to adjust Black Level to subtract this offset from each pixel value.

For adjustment of the Black Level it's recommended to darken the sensor (closed lens) and increase the Black Level value till all pixel values are touching 0 level. The Black Level can vary slighty with changing the sensor clock and should be optimized for each selected speed.

Command:	**:a5<x_1x_0>**	<x_1x_0>: Range, typ. 00h ...ff (hex), typ. 30… 80 hex ASCII char
Response:	ACK	if enabled by separate command :A<n>

6.5 Image size and position

Image size and position within the sensor is defined by four parameters:

Bit(s)	Description
r1[9..0]	Number of first line, 0..$3FD_h$
r3[9..0]	Number of lines, 0..$3FF_h$
r4[6..0]	Address/10 of the first pixel
r5[6..0]	Address/10 of the last pixel

Table 6.5-1

6.5.1 Address of the first line

Register r1 defines the vertical start position of the ROI (region of interest) within the sensor size. It's the first line to be displayed.

Command: **:r1**<$x_2x_1x_0$> <$x_2x_1x_0$> ... Range 000h ...3fdh
Response: ACK if enabled by separate command :A<n>

Example: :r1100 = 256 (dec) + 1 ⇨ image starts at line 257

ⓘ	If dual column binning is activated, r1 is doubled within the camera logic.

6.5.2 Number of lines

Register r3 defines the vertical size of the ROI (region of interest) within the sensor size. It's the number of lines to output.

Command: **:r3**<$x_2x_1x_0$> <$x_2x_1x_0$> ... Range 000 h ...3ffh
Response: ACK if enabled by separate command :A<n>

Example: :r31ff = 511 (dec) +1 ⇨ display 512 lines

ⓘ	The sum of r1 and r3 must be ≤ 0x3ff/1023 or 0x1ff/511 if dual column binning is activated!

6.5.3 Address of the first pixel of a line

Register r4 defines the start of horizontal start position of the ROI (region of interest) within the sensor size. It's the leftmost pixel. The value is the pixel address divided by ten.

Command: **:r4**<$x_2x_1x_0$> <$x_2x_1x_0$> ... Range 000h ...7fh
Response: ACK if enabled by separate command :A<n>

Example: :r4020 = 32 (dec) * 10 ⇨ line start at pixel 320 (dec)

Calculation of the value of r4: Value of r4 = Pixel-Nr./10

6.5.4 Address of the last pixel of a line

Register r4 defines the horizontal end position of the ROI (region of interest) within the sensor size. It's the rightmost pixel. The value is the pixel address divided by ten.

Command:	**:r5<*$x_2x_1x_0$*>**	<$x_2x_1x_0$> ... Range 000h ...07fh
Response:	ACK	if enabled by separate command :A<n>
Example:	:r505f	= 95 (dec) * 10 ⇨ line end at pixel 790 (dec)

Calculation of the value of r5: Value of r5 = Pixel-Nr./10

Calculation of horizontal size: (r5 – r4 + 1) * 10
e.g.: r4 = 010, r5=04f (95 - 32 + 1) * 10 = 640 pixel per line

The difference r5 - r4 must be in the range: 0 ≤ r5-r4 ≤ 7fh .

6.5.5 Tracking

For rapid window movement even at slow baud rates a short command is provided..

command: **:t<*n*>,<*m*>**
<*m*> = X-position in pixel/10, range 00h ...07fh
<*n*> = Y-position in lines / 4, range 00h ...0ffh

Response: ACK if enabled by separate command :A<n>

6.6 Clock selection

The MC132x is equipped with a 2-channel programmable clock synthesizer. One channel controls clock frequency of the sensor (sensorclock, F_{sens}), the other controls the frequency of the pixel clock (pixel-clock, F_{pix}). These independent clocks allow an always optimal ratio depending on the product of (image size x image frequency) and the data rate on the output.

As the sensor outputs 10 pixel per clock a sensor clock of 6.6MHz could be chosen. Because the sensor can run up to a clock frequency of 66 MHz only 1/10 of the sensors possible speed would be used. To make use of the maximum sensor clock and maintaining the maximum data rate on the output, just 120 (1280/10 rounded to steps of 10) from the possible 1280 pixel per line can be selected.

Therefore the ratio of F_{sens} and F_{pix} depends on the selected line length:

$F_{sens} <= (F_{pix} \bullet 1280) / (5 \bullet \text{line length})$

or if 100 pixel line length is chosen:
$F_{sens} = (33 \bullet 1280) / (5 \bullet 100) = 70{,}4$ MHz

As this exceeds the maximum sensor clock frequency, F_{sens} is chosen as 66 MHz and F_{pix} as 33 MHz.

6.6.1 Arbitrary selection of sensor and pixel clock

Sensor and pixel clock can be set to any value, the product of: (sensor clock • line length/1280) must always be smaller (about 10%) than the qoutient: (pixel clock / 2).

Command	**:S <x_0>**	<x_0> ... 6 characters, as described in chapter Frequency selection
Response:	ACK	if enabled by separate command :A<n>

6.6.2 Image Format/Speed change

There are several steps necessary for a change of image format:

i. Disable sensor controller with :r6[4] = 0.
ii. Set image size with (:r1,:r2, :r3, :r4, :r5).
iii. If new sensor clock = old sensor clock:
    1. Do not set pixel clock nor sensor clock.
iv. If new sensor clock > old sensor clock:
    1. Set new pixel clock (:S6…..), then new sensor clock (:S4….).
v. If new sensor clock < old sensor clock:
    1. Set new sensor clock (:S4…..), then new pixel clock (:S6….).
vi. Reenable sensor controller (:r6[4]=1).

6.7 Exposure control

Exposure control is selected with register r6[7..4] and register r2[9..0].

Bit(s)	Description
r6[7..4]	Type of exposure
r2[9..0]	Exposure time

table 6-1

6.7.1 Type of exposure

The MC132x can expose the images synchronous or asynchronous. An external signal on CC1 can be used to synchronize MC132x cameras to each other or to an external event.

6.7.1.1 Synchronous exposure

Synchronous exposure means that the next image is exposed, while the current image is output. This mode provides fastest frame rate while maintaining maximum exposure time as long as 1/frame rate. If an external synchronization signal is input on CC1 its frequency range can be between 30Hz and the selected free running frame rate. Use MC13xx camera configuration tool for selection.

6.7.1.2 Asynchronous exposure

With asynchronous exposure, an external signal starts exposure, and the exposed image is output immediately after the exposure ends. Exposure time is defined either by an internal timer or by the width of the external EXP (CC1) signal. The time between two consecutive EXP (CC1) edges can be indefinite.

Frame rate = 1/(exposure time + image output time). Image output time equals the selected free running frame rate. Use MC13xx camera configuration tool for selection.
The following registers select exposure type:

Register Bits	:r6[7..4]	:r7[8]	:rf[0]
camera stop	xxx0	x	x
Synchronous without electronic shutter	0001	0	0
Synchronous with electronic shutter	0011	0	0
Synchronous with electronic shutter and external synchronisation signal, positive edge	0011	0	1
Synchronous with electronic shutter and external synchronisation signal, negative edge	0011	1	1
Asynchronous, pulsewidth, positive edge	1011	0	0
Asynchronous, pulsewidth, negative edge	1011	1	0
Asynchronous, timer, positive edge	1111	0	0
Asynchronous, timer, negative edge	1111	1	0

Table 6.7-1

6.7.2 Frame rate with synchronous exposure

The frame rate with synchronous exposure is direct proportional to the selected number of lines.
The time for one line is:

$T_{zz} = 1/F_{sens} \bullet 136$ [sec]

T_{zz} ...time/line
F_{sens}... Sensorclock

Frame rate: = 1 / (time/line • number of lines+1) or:
= F_{sens}/ (136 • (r3[9..0] +2))

Dependencies between image size and frame rate for typical clock frequencies are given in the following table:

image size	100x100	240x240	640x480	1280x1024
Sensorclock (MHz)	66	33	13,2	6,6
Time/line [µs]	2,06	4,12	10,3	20,6
Frames/s	4.852	1.011	202	47

Table 6.7-2

6.7.3 Synchronous operation without shutter

Without electronic shutter the exposure time is 1 / frame rate.

6.7.4 Synchronous operation with shutter

In the sensor is implemented a freeze frame shutter, which allows to reduce the exposure time in steps of one line. The minimum value of the exposure time is the duration of 2 line periods, which is determined by the value of r2 (min. 001h).

Command:	**:r2<$x_2x_1x_0$>**	<$x_2x_1x_0$> ... Range 001h ...3ffh
Response:	ACK	if enabled by separate command :A<n>

Exposure time T_B :

$$Tt_B = r2 \bullet T_{ZZ} - T_{ZZ} / 2$$

T_B ... exposure time in s
r2 ... value of register 2
T_{ZZ} ... time/line
$T_{zz} = 1/F_{sens} * 136$ [s]
Tt_{zz} ... Time/line
F_{sens}... sensor clock

Typical exposure times:

Sensor clock frequency (MHz)	Time/line (µsec)	r2 @ 1/5.000 s	r2 @ 1/10.000 s
66	2,06	97	49
33	4,12	49	24
13,2	10,3	19	10
6,6	20,6	10	5

Table 6.7-3

6.7.5 External synchronisation with synchronous exposure

MC132x cameras can be synchronized to an external signal that is input on the EXP (TRIG) signal. The strobe output signal of a MC132x “master camera” can be used for that purpose.

See timing diagram: Synchronous exposure with external synchronisation via TRIG

The cameras frame rate must be set to a frequency slightly higher than the maximum frequency of the synchronization signal. The minimum frequency should be higher than 30Hz.

Command: **:rf001**
000 … deselect external sync signal
001 … select external sync signal
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register .
Response: ACK if enabled by separate command :A<n>

Command: **:rf001**
<0> ... deselect external sync signal
<1> ... select external sync signal
Response: ACK if enabled by separate command :A<n>

Make sure that a sync signal is present on EXP/CC1 before this command is issued or the “trigger on CC1” button is pressed on the MC13xx camera configuration tool.

The polarity of the sync signal can be selected with the _Polarity_of_EXP-signal

6.7.6 Frame rate with asynchronous exposure

The frame rate with asynchronous exposure = Frame rate with synchronous exposure – (1 / exposure time).

6.7.7 Asynchronous exposure, shutter control by pulse width

This operating mode is selected with register 6:
:r6[7..4] = 0xb

Exposure time depends on the width of the external EXP – signal.

6.7.8 Asynchronous exposure, shutter control by timer

This operating mode is selected with register 6:
:r6[7..4] = 0xf

The asynchronous exposure time is dependent on :r2[9..0]. The exposure timer counts as many lines as are defined in register :r2[9..0].

Exposure time:

$T_B = 1/F_{sens} * 136 \bullet (1+r2[9..0])$ [Sec]
T_B ... exposure time
F_{sens}.. sensor clock

example: sensor clock = 66MHz
value of r2[9..0] = 6
$T_B = 136 \bullet 6 \bullet 15$ ns = 12,2 µs

6.8 Extended command features

For conformity to standards, e.g. GigE Vision the command structure of the camera was extended. It supports separate commands for direct input of horizontal position, horizontal size and vertical size.

Command:	**:l<nn><xxxxxxxx>**	direct programming of additional 32bit registers <nn> … no. of register, range 00 … 02 (hex in ascii characters) <xxxxxxxx> … 32bit value of register, hex in ascii characters
Response:	ACK	if enabled by separate command :A<n>

Defined are following registers (command example):

:l00…	horizontal start position, hex value as ascii characters, range (<xxxxxxxx> =): 00000000 … 0000007e (same value as for register r4)

:l01…	horizontal size, hex value as ascii characters, range: 00000000 … 00000080, value of l01 = r5 – r4 + 1
:l02…	vertical size, hex value as ascii characters, range: 00000000 … 00000500 value of l02 = r3 – r1 + 1

New inputs with command :l will only adjust the values of registers r1…rf and will not be stored separately in the camera profile.

6.9 Special fpga registers

The fpga also handles 16 registers, each 32 bit wide, for setting special modes or functions inside the camera. These settings may be coupled with special features, e.g. sobel filter or customized image preprocessing inside the camera, which can be en- or disabled in dependence of camera extensions defined in the identifier.

The 16[th] register is reserved. It can only read and contains the 4 LSB bytes of the identifier.

Command:	**:m<n><xxxxxxxx>**	**<n>** … no. of register, in hex as ascii character, range 0…e (hex) <xxxxxxxx> … register value, in hex as ascii character, MSB…LSB, range 00000000…ffffffff (hex)
Response:	ACK	if enabled by separate command :A<n>
Example:	:m012345678	

The setting of the m registers may change depending on the camera functions. There are no presetting. For standard cameras the registers are preset to 0000000. If used for special features the adjustment is described in the documentation of the special feature(s).

6.10 Firmware

The camera possesses programmable devices, which are working with some firmware packages. New cameras were programmed with all needed firmware packages and will not need any update.

For customized firmware or additional features the camera offers the possibility to update some of the firmware versions. The procedure of updating depends on the firmware package.

Do not update more than 1 firmware at the same time.
In case of updating more than 1 firmware, please start with application program, then fpga program follows.

6.10.1 Update firmware microcontroller application

The microcontroller works with 2 programs, the bootloader and the application program.

The bootloader is the basic program of the microcontroller, which ensures some basic functions (e.g. communication, loading application program) and cannot be changed or updated. In standard use of the camera it will never work in the bootloader program and the version cannot be read out. It's only used for updating the application program.

The application program is the active microcontroller program in the camera, which supports communication, data handling and fpga program updates.

See description of update procedure in chapter “Firmware update procedure”.

6.10.2 Update FPGA firmware

MC132x’s logic is integrated into a FPGA (**F**ield **P**rogrammable **G**ate **A**rray), which’s configuration is stored in an EEPROM. Upon power up or a command the FPGA is loaded with this configuration. Configuration data can be downloaded via the serial interface. Mikrotron may provide configuration files (*.ibf) on request.

After download of configuration data, this data is permanently stored in EEPROM and the FPGA is configured with the new data. Besides a power cycle, the **:c** command can be used to reconfigure the FPGA with the internally stored configuration data.

See description of update procedure in chapter “Firmware update procedure”.

6.10.3 Firmware update procedure

Before you disable the loaded firmware please ensure that you have a adequate application firmware version to load (e.g. MC132x….A202.ibf for application firmware, MC132x…F070.ibf for fpga firmware).

(1) Start camera and test communication, e.g. with tool press “info camera” and wait for response (serial no. and firmware)

(2) Select in menu “Write” “Write string to camera” and write command
- for application firmware update: :ERASEAPP <ENTER>
- for fpga firmware update: :ERASEEPCS1 <ENTER>

which erases the program and for application firmware will restart the camera.

After this command the camera may not be able to deliver any images, load/send/store register or profile data. The status led of the camera will turn to red.

(3) For application firmware update only:
The camera now starts with the bootloader program. It is displayed during start by the status led which blink one time during power up.
Test communication, e.g. with MC13xx in menu “Write string to camera” and command :v.
Response (e.g.): B2.02

(4) Select in menu “Write” “Write file to camera” and choose application file MC132x*.ibf) and open it. The file transfer will start immediately. If the camera recognizes a newfirmware it will switch off the status led.

Download of *.ibf file via serial link takes app. 1,5 - 2 min depending on the used camera. There should be no loss of power or communication during this time!

(5) Wait until file transfer is finished and the status led turns on. If the upload of the file was successful, the led will turn to green, otherwise it will be red.

(6) Verify version string by reading serial no. and firmware versions (command: :v). The new firmware version will be displayed as part of the version string. If the version is identical to the expected the camera is ready to use for capturing images.

6.11 Horizontal pixelbinning

Pixelbinning adds the gray values of two adjacent pixels and outputs it as one pixel with double sensitivity. In X-direction only 512 pixels are needed to cover the sensors full size.

To retain aspect ratio, every second line is discarded, if this feature is not disabled by setting Bit 8 of register 6 **(:r61xx)** or vertical pixelbinning is activated.

Command: **:r7010**
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register .

Response: ACK if enabled by separate command :A<n>

If discarding of every other line is not disabled **(:r61xx)**, the contents of **:r1** is doubled in camera logic. To address a specific line on the sensor, the value written into **:r1** has to be divided by two and **:r3** must not exceed 1ffh.

Example:
To output 256 lines from line 128, set r1 = 63 and r3 = 255 (=0xff).

6.12 Vertical pixelbinning

Vertical pixelbinning adds the gray values of two superimposed pixel of a column. This doubles sensitivity and vertical field of view. To retain aspect ratio, in addition horizontal binning must be activated. To activate, set bit 2 in register 6.

Command: **:r6034**
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register .

Response: ACK if enabled by separate command :A<n>

6.13 Digital gain

Out of the 10-bits sensor data either the most significant 8 bits (gain 1), or bits 8..1 (gain 2), or the least significant 8 bits (gain 4) are selected.

Command: **:r700x** x = 0: gain 1
x = 4: gain 2
x = 8: gain 4
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register.

Response: ACK if enabled by separate command :A<n>

6.14 External clock input

MC132x frequency synthesizer can use the Camera Link® used to synchronize several MC132x to one master clock.
To activate set Bit 9 of register 7.

Command: **:r7200** :r7[9] = 0, external clock input disabled
:r7[9] = 1, external clock input enabled
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register.

Response: ACK if enabled by separate command :A<n>

If the external reference clock is different from 3.6864 MHz, the codes for the clock synthesizer have to be recalculated.

6.15 Polarity of EXP-signal

The polarity of the EXP-signal can be positive- or negative active. Use :r7[8] to select.

Command: **:r7100** :r7[8] = 1, negative polarity
:r7[8] = 0, positive polarity
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register.

Response: ACK if enabled by separate command :A<n>

6.16 Test image

For testing of camera logic and video data transmission, sensor data can be replaced by an internal gray scale pattern with pixel values of 0..127. Use digital gain command to see pixel values of 0..255.

Command: **:r7040** r7[6] = 0, internal gray scale pattern disabled
r7[6] = 1, internal gray scale pattern enabled
This is an example for enabling this function. Depending on the wanted camera mode it can be necessary to add additional bits of this register.

Response: ACK if enabled by separate command :A<n>

6.17 Image counter

If a sequence of frames is to be recorded for long time at a high frame rate, it can be useful to mark the images for later identification or check for completeness.
MC132x has a 16-Bit image counter whose count can replace the first two pixel of every image. The image counter is cleared with every low to high transition of r7[1], the camera enable bit. It is incremented by every new image.

Command example: **:r7002** r7[1]

Response: none

6.18 ImageBLITZ® shutter release

ImageBLITZ can replace an external signal (e.g.: a light barrier) to release the shutter. Like a light barrier, ImageBLITZ is used to capture fast moving objects on the exact same position on the image.

Contrary to the light barrier, ImageBLITZ uses the same information as condition to release the shutter as the then exposed image.

ImageBLITZ defines one specific line or a part of the 1024 lines as trigger window. This is true even if the selected image size is less 1024 lines or outside of the selected image area.

After activation of ImageBLITZ and after issuing the EXP signal as an enable signal, the MC131x hardware checks the gray values in the trigger window at a repetition rate that is defined by the exposure time selected with bits 3..0 of r6.

If a selectable number of pixels along that trigger window exceed or fall short of a selectable threshold, one single image is exposed and output.

To adjust ImageBLITZ®, the trigger line can be superimposed to the image. Within the selected line, 10 pixel are displayed as a dotted black- and white line as long as the selected threshold is not passed.

ImageBLITZ is configured with the registers r8..rC_h:

6.18.1 ImageBLITZ® processing

When ImageBLITZ® is activated with $:r7_h[0] = 1$:

1. Wait for an active edge on the EXP input.
2. The MC13xx exposes a line, that was chosen with :rC[9..0] and is called trigger line, for an exposure time defined by :r2[9..0]. It compares the intensity of a group of 10 pixel along the selected trigger line against an adjustable threshold ($:rA_h[7..0]$, Range: 255..0).
3. The number of exceedings ($:rA_h[8] = 0$) or fall backs ($:rA_h[8] = 1$), are counted, and the result is compared to a second threshold ($:rB_h[6..0]$, Range: 127..0).
4. Each time this threshold is exceeded (release condition); an “inhibit counter” ($:rD_h[9..0]$, Range 1..255) is loaded.
5. The inhibit counter” $:rD_h[9..0]$ is counted down, each time the “release condition” was not reached. Once this “inhibit counter” has expired, a new image is exposed and output.After image is output, repeat at 1.

6.18.2 ImageBLITZ® programming

ImageBLITZ® is programmed by registers r8..rD_h and activated with r7[0].

6.18.2.1 Address of trigger line

The register rC_h determines the vertical position of the trigger line in the image.

command: **:rC_h <$x_2x_1x_0$>**

<$x_2x_1x_0$> ... range 00h ...3ffh

Response: none

Example: :rc100
100h = 256

ⓘ	In pixelbinning mode the value of rC is internally doubled. The value must not be higher than 1ffh/511.

6.18.2.2 Leftmost pixel of the trigger line

The value of register r8 / 10 is the number of the leftmost pixel in the trigger line.

Command: **:r8<$x_2x_1x_0$>**
<$x_2x_1x_0$> ... range 000h ...07fh
Response: none

Calculation of r8:
Value of r8 = pixel number / 10

6.18.2.3 Rightmost pixel of the trigger line

The end of the trigger line is determined by the value of register r9.

Command: **:r9<$x_2x_1x_0$>**
<$x_2x_1x_0$> ... range 000h ...7fh
Response: none

Calculation of r9:
Value of r9 = pixel number / 10

6.18.2.4 Threshold level, mark trigger line

The threshold level is set by register rA_h . The pixel values along the trigger line are compared with this value.

Command: **:rA_h <$x_2x_1x_0$>**
<x_1x_0> ... range 0 ..ffh
<x_2> = 0: pixel gray level > threshold level, trigger line not visible
1: pixel gray level < threshold level, trigger line not visible
2: pixel gray level > threshold level, trigger line visible
3: pixel gray level < threshold level,

trigger line visible

Response: none

The trigger line is displayed as dashed, black and white line. One dash has a length of 10 pixel. The trigger line is only displayed in parts of the line where the pixel fulfill the trigger requirements.
Under normal operation conditions the trigger line will be visible only in parts. The number of dashes may be counted and used for the setting of register rB_h.

6.18.2.5 Release condition

Register rB_h contains the release condition.
The release condition is determined by the number of pixels along the triggerline that fulfill the trigger requirements.

Command: **:rB_h $<x_{9..0}>$**
$<x_{6..0}>$ = 0 ..7fh, number of pixel that match the trigger requirements
$<x_{8..7}$ = 0: correction value 0 for the X - position of output window
$<x_{8..7}$ = 1: correction value 4 for the X - position of output window
$<x_{8..7}$ = 2: correction value 8 for the X - position of output window
$<x_{8..7}$ = 3: correction value 12 for the X - position of output window

Response: none

6.18.2.6 Release Inhibit

The Release Inhibit function is defined with :rD_h. It tells ImageBLITZ how often sequentially the “release condition” must **not** be met, before an image is output.
This feature allows to trigger an object only once on the dark- to bright edge of the scene. This avoids retriggering, once the trigger condition was met and the object is still visible within the triggerline after the image has been output.

Command: **:rD_h $<x_{7..0}>$**
$<x_{7..0}>$ = 0 ..ffh, number of fulfilled, sequentially trigger conditions

Response: none

6.18.3 ImageBLITZ® registers

Register	Bit	Description
r7	0	= 1: activate ImageBLITZ®
r8	6..0	First pixel mod. 10
r9	6..0	Last pixel mod. 10
rA_h	7..0 8 9	Exposure threshold 1: bright object triggers 0: dark object triggers 1: make triggerline visible
rB_h	6..0 8..7 9	Number of exceedings or fall backs, release condition, X – tracking correction X – tracking enable.
rC_h	9..0	Address of triggerline
rD_h	7..0	exposure limitation, number of exposures without exposure condition until an image is captured

Table 6.18-1

Registers r1..r7 are programmed according to image size and position and for Asynchronous operation, timer .

Register	Bit	Description
r1, r3..r5		Image size and position
r2 r6	9..0 7..4	Async operation, timer 0fh

Table 6.18-2

6.18.4 ImageBLITZ® setup

The MC131x is configured for asynchronous operation with timer , registers r8, r9 and rC_h are loaded for the desired position of the trigger line. Register rB_h is loaded with 0, register rA_h with 201_h, so that the trigger line is visible.

ⓘ	If the image is zoomed down for display by an application program, every other line may be omitted and the trigger line may then disappear.

ImageBLITZ® is enabled with Register r7 Bit1=1.

Now position the trigger line with the registers r8, r9 and rC_h across the object that is used for the shutter release..

Clear Bit 8 in Register rA_h if a bright objects releases the shutter, set rA_h[8] if dark objects release the shutter. While the trigger line is placed across the object, raise threshold with rA_h[7..0] until as many dashes from the trigger line disappear as are loaded in Register rB_h [6..0]. This is called the release condition.
If it is expected that the release condition is met more than once for a single object, load rB_h [9..7] with a number of exposed lines that will not met the release condition before exposing one image.

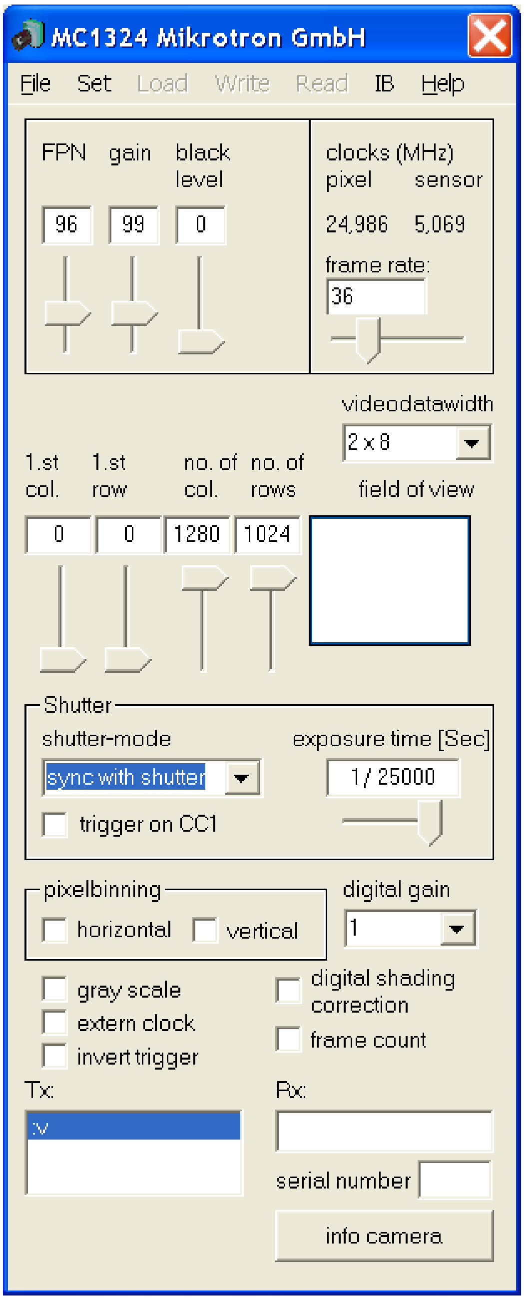
7 MC13xx configuration tool

The MC13xx configuration tool must be installed on a Windows PC. (Win9x, WinNT, Win2K, WinXP) by means of the setup software. See also www.mikrotron.de to download the latest version.

This software provides an almost self explaining user interface to modify any camera parameter. The description of the parameters follows the marked chapters in this user manual.

To use this tool with the camera MC13xx the serial interface is integrated in the Camera Link® interface. You do not need any other additional cable.

7.1 Basic Configuration



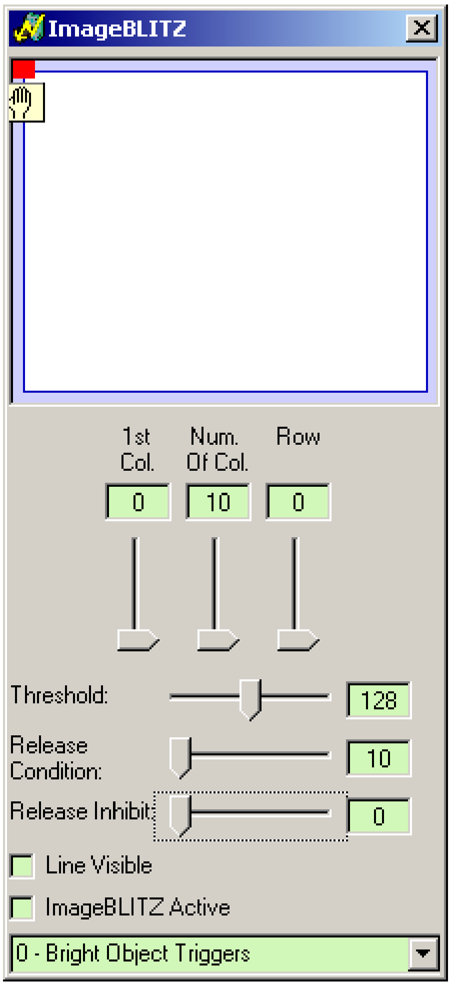
File: Save or read settings to or from file.
Set: Select com port. If Inspecta-4D and the correct cable is used, the MC13xx can be written to but not being read from.

Load, Write, Read: Profile processing

FPN, Gain, black level: Adjusting image
Clocks, frame rate: Clock selection
1.st col…num. of rows: Adjusting Image
Shutter: Type of exposure
Frame count (6.17),
gray scale (6.16),
invert trigger (6.15),
extern clock (6.14),
digital gain (6.13)
pixelbinning……(6.10)

Info camera:
Read serial number and firmware version
Tx:
Display control strings
Rx:
Display response

7.2 ImageBLITZ® Configuration



1st Col, Num Of Col., Row (Position of TriggerLine):
r8, r9 and rC

Threshold:
rA

Release Condition:
rB

Release Inhibit:
rD

Line Visible:
rA Bit 8

ImageBLITZ Active:
r7 Bit 0

Bright Object Triggers:
rA Bit 9

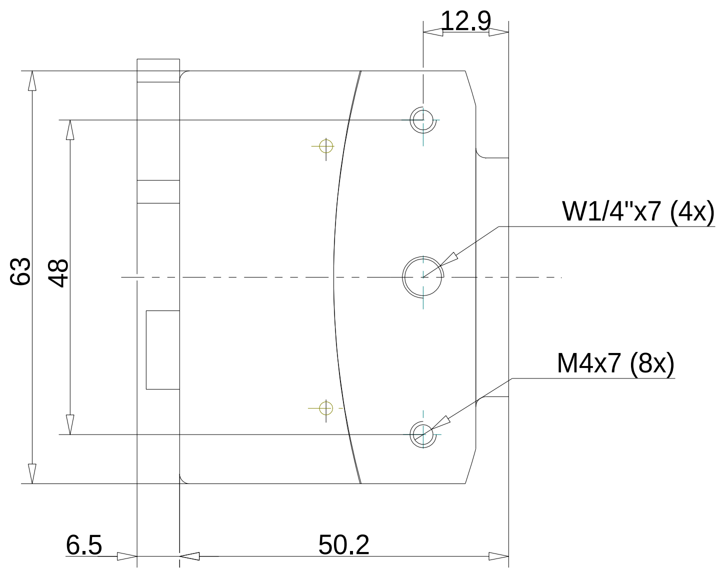
8 Mechanical dimensions

8.1 Camera body MC132x

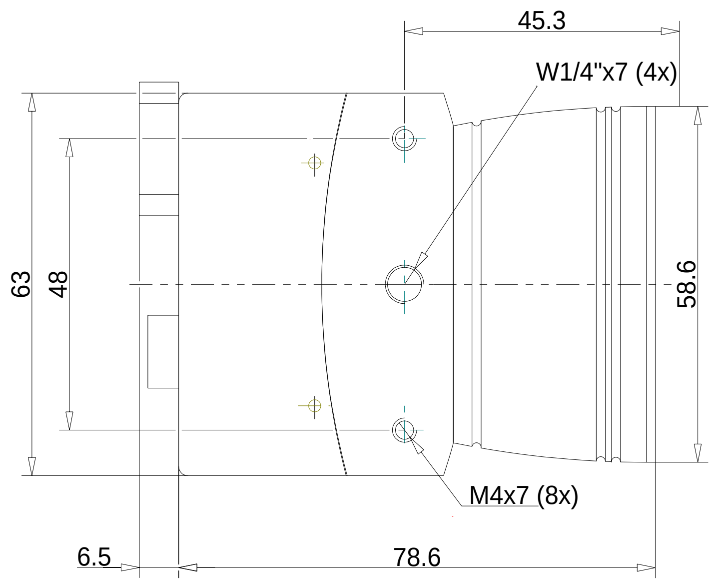
The camera body has (without lens) has very compact. To fasten the camera there are two mounting holes M4x7mm and one tripod connection on each side available.

8.1.1 Dimensioned drawing, side view of MC1324/25

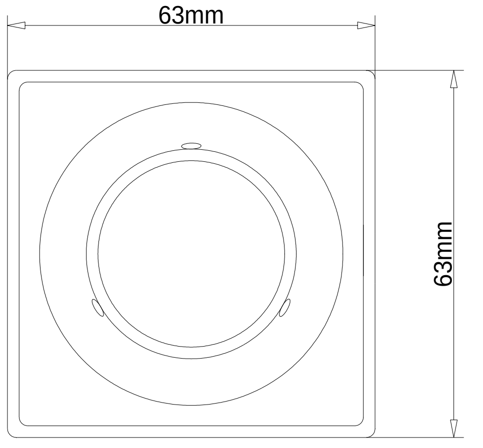
all dimensions im mm



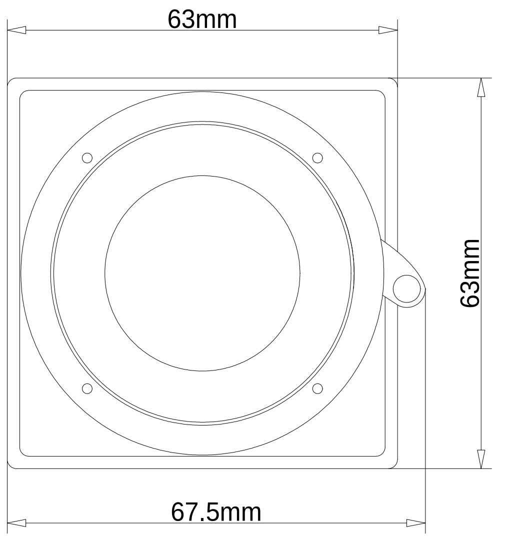
8.1.2 Dimensioned drawing, side view of MC1326/27



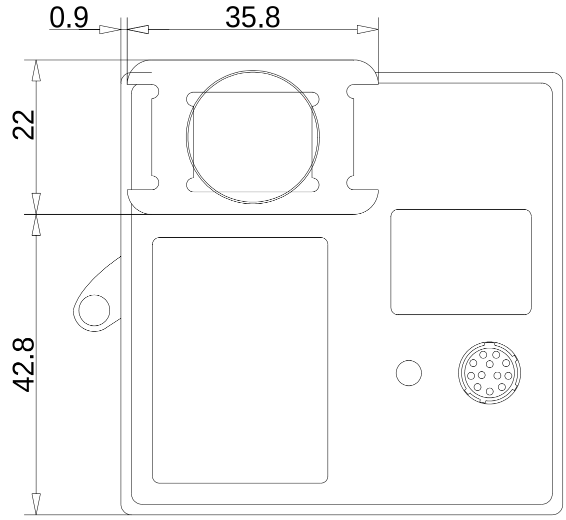
8.1.3 Dimensioned drawing, front view of MC1324/25



8.1.4 Dimensioned drawing, front view of MC1326/27



8.1.5 Dimensioned drawing, rear view of MC132x



8.2 Lens adjustment

8.2.1 Lens adaption

In dependence of the camera model the MC132x camera are prepared for either C-mount or F-mount connection (see table overview of cameras).

Camera types with F-mount adaption may be converted to C-mount adaption by exchanging the lens mount flange of the camera, which is attached with 4 screws. The C-mount lens flange is not scope of delivery of the F-mount camera types and is offered separately.

8.2.2 Adjustable lens adapter, only for camera models with C-mount flange

For fine adjustment of the focal length a lens adapter with an adjustment range of ± 1 mm is provided. Use the three screws nearby the sensor window to fasten the lens adapter after a proper adjustment together with the chosen lens.

8.2.3 Lens selection for camera types with C-mount flange (see table overview of cameras)

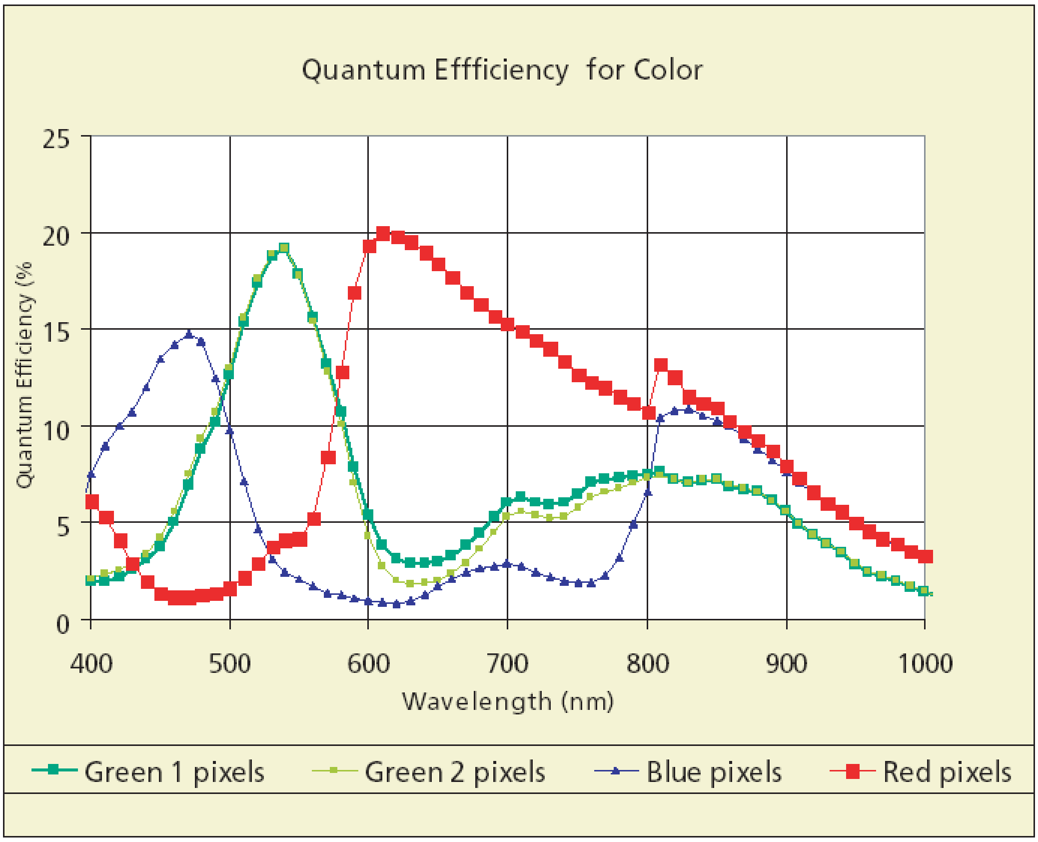
Due to the size of the sensor use C-Mount lenses with the largest possible optical diameter or an adapter for lenses like F-Mount, especially for lenses with a focal length < 25mm..

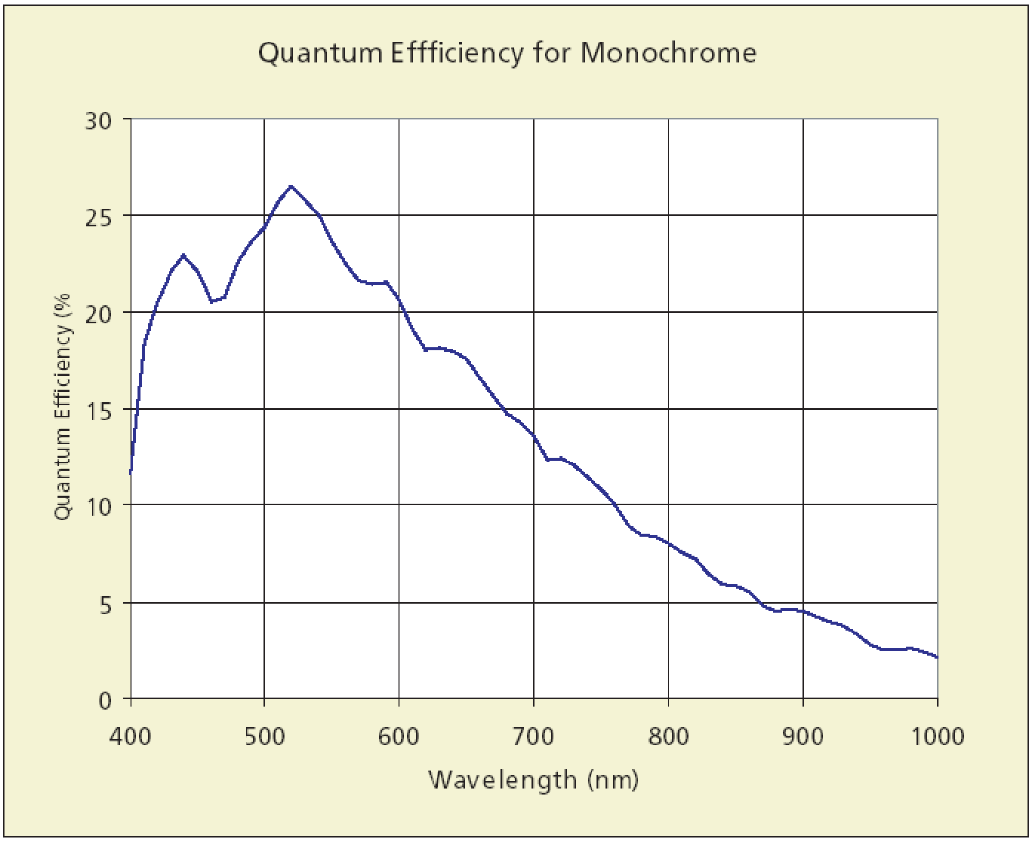
9 Technical Data

MC1324 MC1326 MC1325 MC1327	Monochrome Bayer Filter
Number of pixel	1280 x 1024
Pixel size	12 x 12 µm
Active area	15,36 (H) x 12,29 (V) mm
Fill factor	40%
Sensitivity at 550 nm @ Vref = 1V (a2 = 66h)	1600LSB/lux-sec
Spectral response	400..800nm
Shutter	Electronic „Freeze Frame“ Shutter
Trigger	Asynchronous shutter, shutter time selectable with internal timer or by pulse width of trigger signal
Internal Dynamic	59 dB
Power supply	8 ... 24 V
Power consumption max. Thermal resistance typ.	5 W 0.17°/W
Serial data link	9,6 – 115 KBd, 8 bits, 1 stop bit, no parity
Digital video	GigE Vision® Interface Pleora GigE IPort Interface
Case temperature	+5..50°C
Shock & vibration	70g, 7grms
Dimensions (WxHxD)	63 x 64,7 x 56,27 mm
Case temperature	+5 ... +50° C
Weight	ca. 300 g
Lens mount MC1324/25 MC1326/27	 C-Mount F-Mount

Table 8.2-1

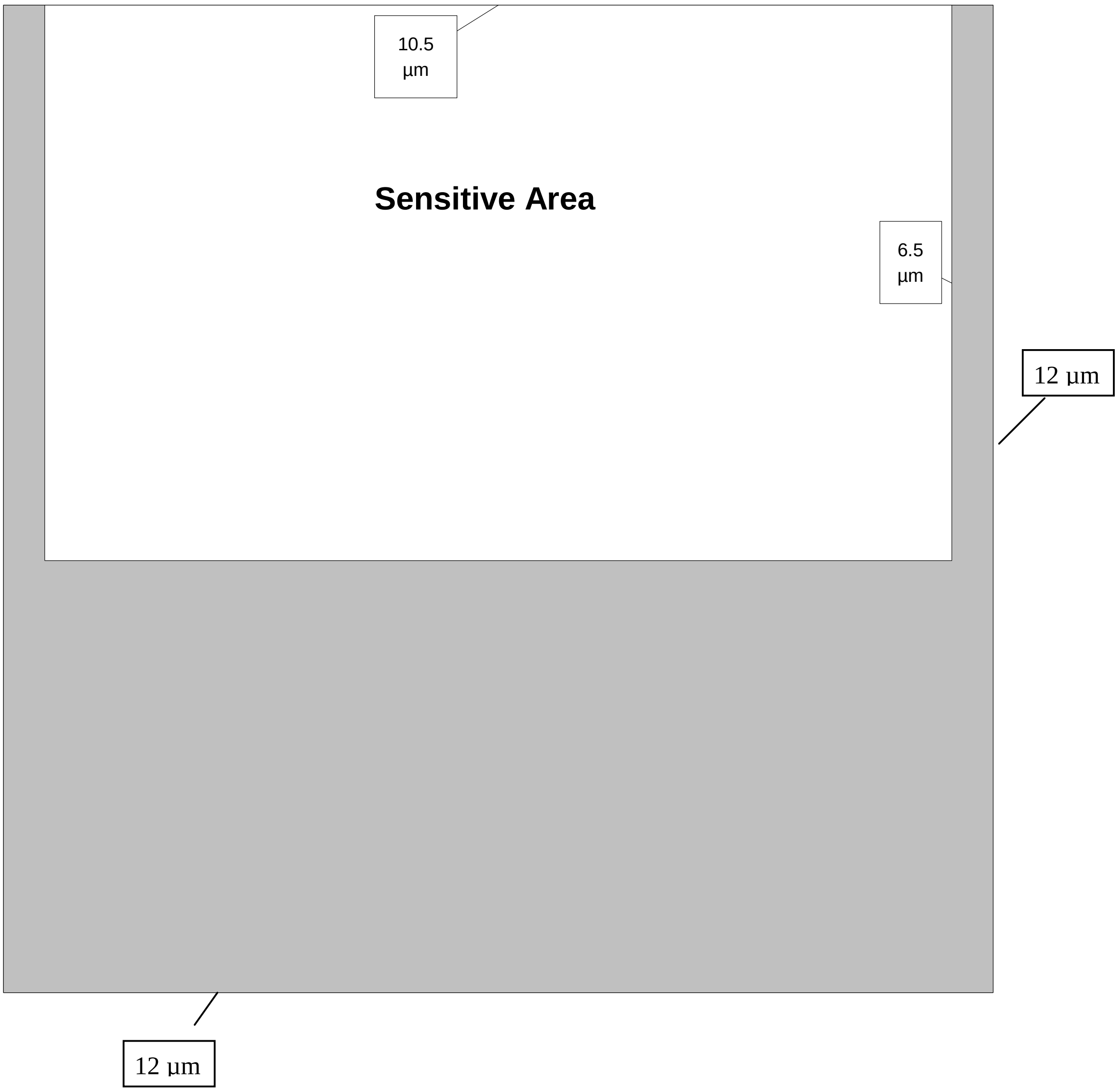
9.1 Spectral response





9.2 Sensitive area of a pixel

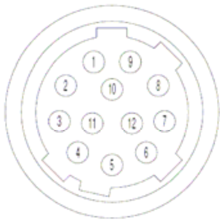
Pixel size:	12 x 12 µm
Fill factor:	40 %
Sensitive area:	10,5 x 6,5 µm



9.3 Connector pinning

9.3.1 Gigabit Ethernet Connector

9.3.2 Circular power connector, 12-pin



pin	signal
1	GND
2	VCC
3	STROBE_GND
4	STROBE
5	TRIG_GND
6	TRIG

pin	signal
7	
8	
9	
10	
11	VCC
12	GND

Table 9.3-1

Manufacturer: Hirose
Order no.: HR10A-10R-12P

Before applying power to the camera we strongly recommend to verify the used pins of the power connector, the polarity (+/-) of the leads and the supply voltage.

The camera may only be used with a supply voltage according to the camera specification. Connecting a lower or higher supply voltage, AC voltage, reversal polarity or using wrong pins of the power connector may damage the camera. If doing so, the warranty will expire immediately.

9.4 Frequency selection

Depending on the selected line length and the datarate of the GigE interface the frequency selection table can provide an optimal ratio of sensor /pixel clock. The pixel clock is only dependent on the selected step and not on the linelength.

The sensor clock is dependent on both the selected step, the line length and the data width. The tables show the selectable frequencies and the correponding codes to program the synthesizer accordingly.

9.4.1 Table of clock frequency codes

For free programming of the clock frequencies of pixel and sensor clock the following table can be used. All codes in the table show the code for sensor clock. The code for pixel clock is the same but starts with “6”.

Example: Adjustment 1 MHz
Command for sensor clock: :S41bb0b
Command for pixel clock: :S61bb0b

	Pixel and sensor clock must be adjusted in dependence of the used camera mode as described in chapter frequency selection. Undefined settings may lead to functional faults or can damage the camera.

Frequency / MHz		Code	Frequency / MHz		Code	Frequency / MHz		Code
Wanted	Real		Wanted	Real		Wanted	Real	
1.0	1.001	41bb0b	24.5	24.488	416905	48.0	47.923	409081
1.5	1.497	409301	25.0	24.986	41dd07	48.5	48.538	413084
2.0	2.002	41ba8b	25.5	25.498	41408a	49.0	48.976	416885
2.5	2.501	412685	26.0	26.010	41f090	49.5	49.503	416c85
3.0	2.995	409281	26.5	26.496	41c08e	50.0	49.971	41dc87
3.5	3.502	408e03	27.0	27.034	404c81	50.5	50.475	41580b
4.0	4.005	41ba0b	27.5	27.506	41788b	51.0	50.995	41400a
4.5	4.501	41f20b	28.0	28.017	408c83	51.5	51.610	404801
5.0	5.003	412605	28.5	28.508	41c48d	52.0	52.019	41f010
5.5	5.483	41d208	29.0	29.000	41cc8d	52.5	52.477	41d80f
6.0	5.990	409201	29.5	29.491	405481	53.0	52.992	41c00e
6.5	6.502	41f190	30.0	29.983	41dc8d	53.5	53.453	406802
7.0	7.004	408d83	30.5	30.497	416089	54.0	54.067	404c01
7.5	7.495	41dd8d	31.0	31.027	41888a	54.5	54.445	41740b
8.0	8.010	41b98b	31.5	31.502	416c89	55.0	55.012	41780b
8.5	8.499	414187	32.0	32.043	41b88b	55.5	55.513	41f40f
9.0	9.003	41f18b	32.5	32.507	417889	56.0	56.033	408c03
9.5	9.492	419188	33.0	32.914	41e88c	56.5	56.525	405001
10.0	10.006	412585	33.5	33.513	418489	57.0	57.016	41c40d
10.5	10.506	40d983	34.0	33.997	414087	57.5	57.508	409003
11.0	10.967	41d188	34.5	34.518	419089	58.0	57.999	41cc0d
11.5	11.520	40bd82	35.0	35.021	408c82	58.5	58.491	41d00d
12.0	11.981	409181	35.5	35.482	412886	59.0	58.982	405401
12.5	12.493	41dd87	36.0	36.013	41f08b	59.5	59.509	41b80c
13.0	13.005	41f110	36.5	36.495	418088	60.0	59.965	41dc0d
13.5	13.517	404d01	37.0	36.864	406c81	60.5	60.457	409803
14.0	14.008	408d03	37.5	37.478	40e884	61.0	60.993	416009
14.5	14.500	41cd0d	38.0	37.970	419088	61.5	61.440	405801
15.0	14.991	41dd0d	38.5	38.502	416c87	62.0	62.054	41880a
15.5	15.514	41890a	39.0	39.014	41f08a	62.5	62.423	41f00d
16.0	16.022	41b90b	39.5	39.497	412085	63.0	63.004	416c09
16.5	16.457	41e90c	40.0	40.024	412485	63.5	63.520	41b40b
17.0	16.998	414107	40.5	40.550	407881	64.0	64.087	41b80b
17.5	17.510	408d02	41.0	41.011	415886	64.5	64.512	408002
18.0	18.007	41f10b	41.5	41.472	40a882	65.0	65.015	417809
18.5	18.432	406d01	42.0	42.025	40d883	65.5	65.536	413407
19.0	18.985	419108	42.5	42.561	41f089	66.0	65.829	41e80c
19.5	19.507	41f10a	43.0	43.008	408081	66.5	66.355	406001
20.0	20.012	412505	43.5	43.500	40e083	67.0	67.025	418409
20.5	20.506	415906	44.0	43.868	41d088	67.5	67.489	41d00b
21.0	21.012	40d903	44.5	44.605	41d888	68.0	67.994	414007
21.5	21.504	408101	45.0	44.974	40e883	68.5	68.462	40f805
22.0	21.934	41d108	45.5	45.466	408881	69.0	69.036	419009
22.5	22.487	40e903	46.0	46.080	40bc82	69.5	69.515	40fc05
23.0	23.040	40bd02	46.5	46.541	418886	70.0	70.042	408c02
23.5	23.501	40c102	47.0	47.002	40c082			
24.0	23.962	409101	47.5	47.514	41c487			

9.5 Programming sequence, factory profile f3

Example:	resolution:	1.280 x 1.024 pixel
	frame rate:	47 fps
	pixel clock:	35,3 MHz sensor clock: 6,65 MHz
	shutter:	full frame , exposure time: 21 ms

Strings:	:a16d	
	:a277	
	:a34a	
	:a4c8	
	:a5xx	xx... may be any value 00h ... ffh
	:a600	
	:a76a	
	:a81c	
	:r6000	
	:r1000	
	:r23ff	
	:r33ff	
	:r4000	
	:r507f	
	:r7000	
	:r8000	
	:r9000	
	:ra000	
	:rb000	
	:rc000	
	:rd000	
	:re000	
	:rf000	
	:S61ac87	
	:S41878c	
	:r6030	